the globus alliance
www.globus.org

PLEASE DO THIS

- right click on the desktop and open a terminal
- cd ~
- mkdir admin
- cd admin
- wget http://www-unix.globus.org/ftppub/gt3/3.9/3.9.4/gt3.9.4-all-source-installer.tar.gz

GT4 GridFTP for Admins: The New GridFTP Server

Bill Allcock, ANL

NeSC, Edinburgh, Scotland

Jan 27-28, 2005

Outline

- Quick Class Survey
- Basic Definitions
- GridFTP Overview
- Configuring GSI
- Server Configuration
- Running the Server as a user

Administrivia

- Bathrooms
- source ~/.alias
- gt
- gl
- NOTE $GLOBUS_LOCATION
  - critical. EVERYTHING depends on this
  - you can use this to point to different installs
- NOTE: The pre-installed servers are not 3.9.4. logging is not working in 3.9.4 and this is fixed version from CVS

Running the Server as a User the Prelude

- In a shell, do the following:
  - cd ~
  - mkdir admin
  - wget http://www-unix.globus.org/ftppub/gt3/3.9/3.9.4/gt3.9.4-all-source-installer.tar.gz
  - gunzip *.tar.gz
  - tar –xvf *.tar
  - cd gt3.9.4
  - configure --prefix=/home/nesc/admin/gridftp --with-flavor=gcc32dbg
  - make prewsgridftp postinstall
  - You just built GridFTP

Quick Class Survey

- By show of hands, how many…
  - Know what GridFTP is?
  - Can describe the difference between a client and a server (for GridFTP)?
  - Know the difference between a control channel and a data channel?
  - Have used globus-url-copy before?
  - Know what a bandwidth delay product is?
  - install their own software on Linux? (duh)
  - For my info
    - have good tools for monitoring log files

Basic Definitions

Basic Definitions

- Command – Response Protocol
  - A client can only send one command and then must wait for a "Finished response" before sending another
  - GridFTP and FTP fall into this category
- Client
  - Sends commands and receives responses
- Server
  - Receives commands and sends responses
  - Implies it is listening on a port somewhere

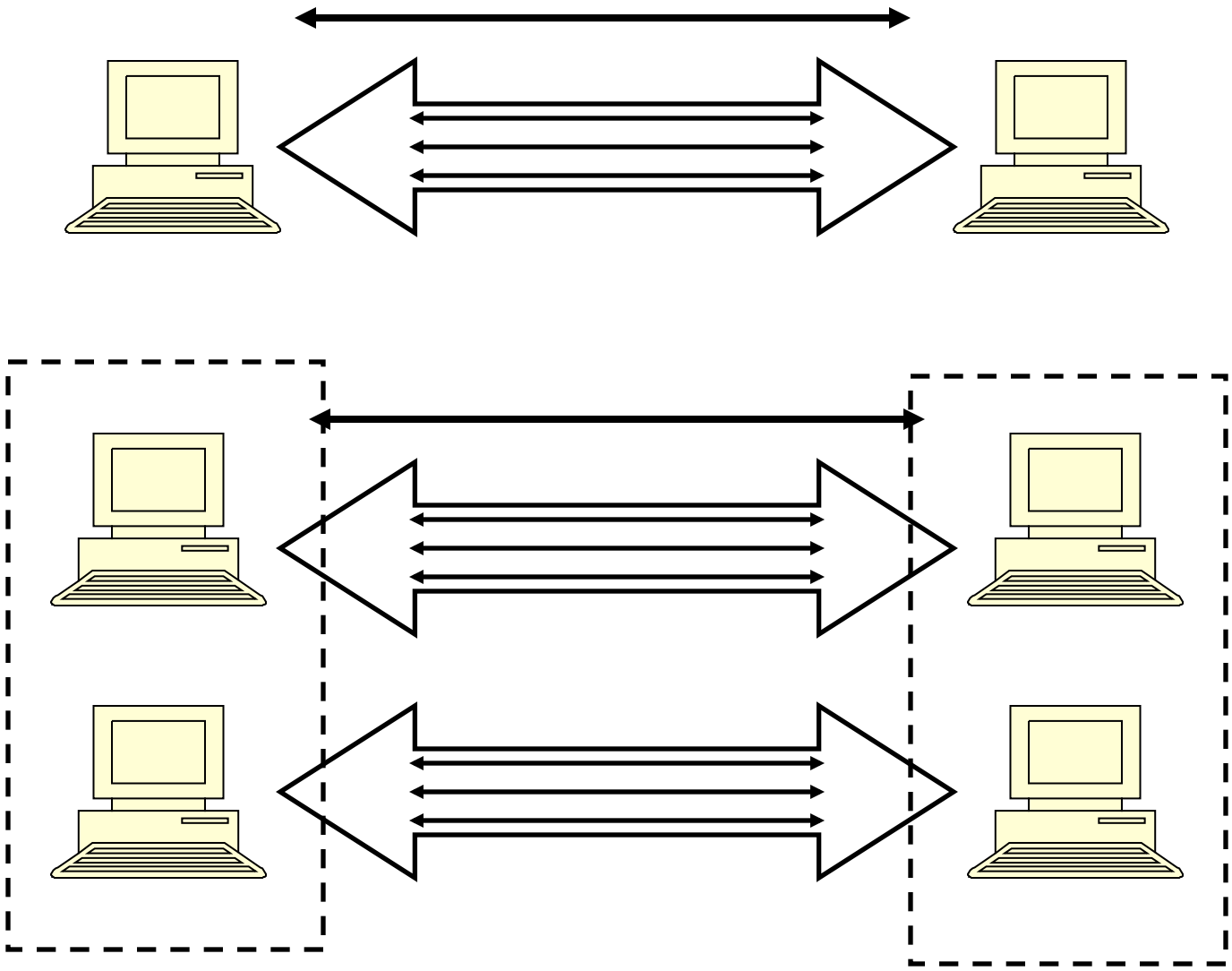
Basic Definitions

- Control Channel
  - Communication link (TCP) over which commands and responses flow
  - Low bandwidth; encrypted and integrity protected by default
- Data Channel
  - Communication link(s) over which the actual data of interest flows
  - High Bandwidth; authenticated by default; encryption and integrity protection optional

Basic Definitions

- Network Endpoint
  - Something that is addressable over the network (i.e. IP:Port). Generally a NIC
  - multi-homed hosts
  - multiple stripes on a single host (testing)
- Parallelism
  - multiple TCP Streams between two network endpoints
- Striping
  - Multiple pairs of network endpoints participating in a single logical transfer (i.e. only one control channel connection)

Parallelism vs Striping



New Server Architecture

- GridFTP (and normal FTP) use (at least) two separate socket connections:
  - A control channel for carrying the commands and responses
  - A Data Channel for actually moving the data
- Control Channel and Data Channel can be (optionally) completely separate processes.
- A single Control Channel can have multiple data channels behind it.
  - This is how a striped server works.
  - In the future we would like to have a load balancing proxy server work with this.

New Server Architecture

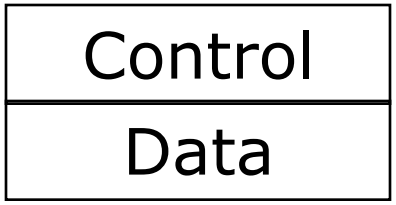
- Data Transport Process (Data Channel) is architecturally, 3 distinct pieces:
  - The protocol handler. This part talks to the network and understands the data channel protocol
  - The Data Storage Interface (DSI). A well defined API that may be re-implemented to access things other than POSIX filesystems
  - ERET/ESTO processing. Ability to manipulate the data prior to transmission.
    - currently handled via the DSI
    - In V4.2 we to support XIO drivers as modules and chaining
- Working with several groups to on custom DSIs
  - LANL / IBM for HPSS
  - UWis / Condor for NeST
  - SDSC for SRB

Deployment Scenario under Consideration

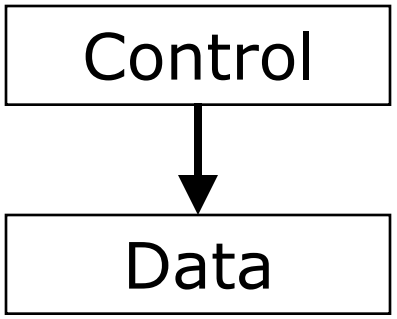
- All deployments are striped, i.e. separate processed for control and data channel.
- Control channel runs as a user who can only read and execute executable, config, etc. It can write delegated credentials.
- Data channel is a root setuid process
  - Outside user never connects to it.
  - If anything other than a valid authentication occurs it drops the connection
  - It can be locked down to only accept connections from the control channel machine IP
  - First action after successful authentication is setuid

Possible Configurations

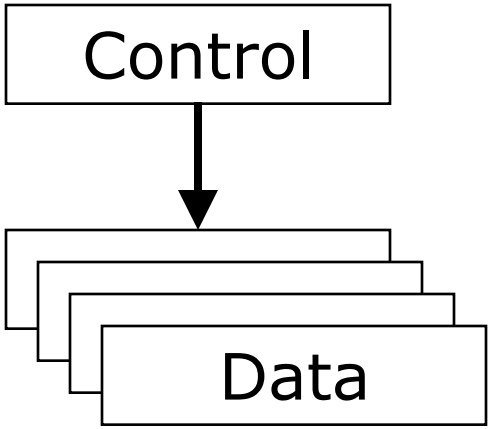
Typical Installation



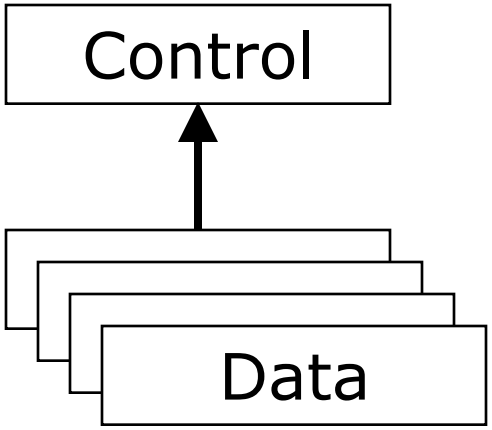
Separate Processes



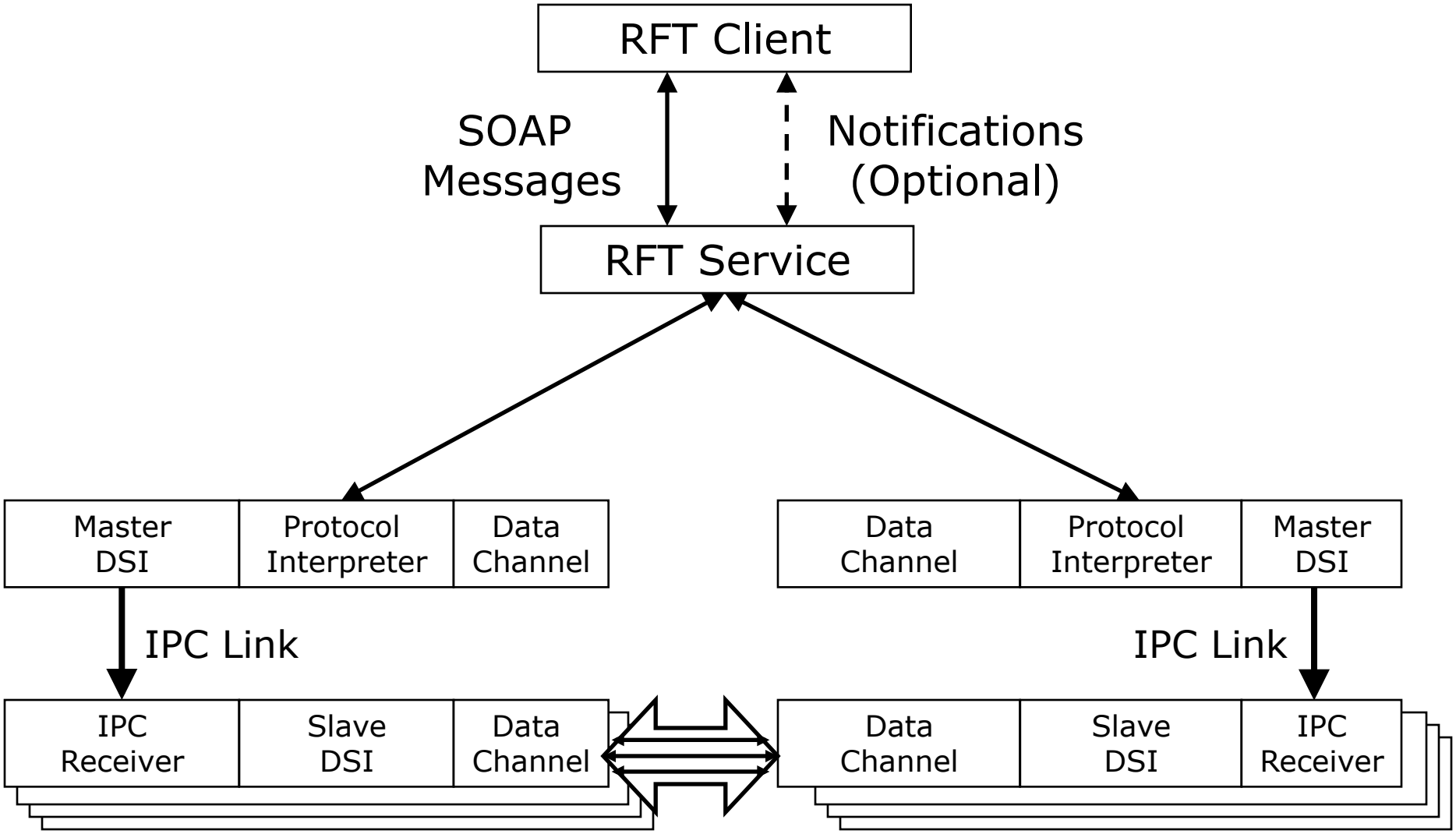
Striped Server



Striped Server (future)



Third Party Transfer



Configuring GSI

Security is a major part of setup

- Likely, the vast majority of the problems you have will be related to security setup.
- The good news is, once it configured correctly, it should just sit there and run.

What the hell is a certificate anyway?

- All things in Globus do "mutual authentication"
  - both sides have to prove who they are
- The magic of PKI lies in the fact that you get given a key and a cert. They have the property that something encrypted with the key can only be unecrypted with the cert and vice versa.
- You keep your key secret, your cert, you pass far and wide (like the CA cert)
- To know if the person on the other end is who they say they are, encrypt something with their public key, if they send it back unencrypted, you know they are in possession of the private key

The pieces of the puzzle (Security)

- User Setup
  - ~/.globus/usercert.pem(user rw, world read)
  - ~/.globus/userkey.pem (user read only)
  - Permissions are critical. It will not work if they are wrong and the errors can be far from obvious
  - The certificate has contains a Distinguished Name (DN). This essentially is your identity on the grid.
  - grid-cert-info -subject

The pieces of the puzzle (Security)

- Host Setup
  - /etc/grid-security/hostcert.pem (root rw, world read)
  - /etc/grid-security/hostkey.pem (root real only)
  - The hostcert also has a DN. The Common Name part (the part you control) needs to be the fully qualified domain name of the host (lab-02.nesc.ed.ac.uk, not lab-02)
  - The client expects the CN to match what a reverse DNS lookup returns.

The pieces of the puzzle (Security)

- Trusted Certificate Authorities
  - /etc/grid-security/certificates/<hash>
  - /etc/grid-security/certificates/<hash>.signing-policy
- grid-mapfile
  - When someone authenticates, you have the subject name of their certificate
  - the grid-mapfile maps this DN to a local user account.
  - This is how admission control is done. Even if they can authenticate (all they need is a cert from a CA you trust), they can't run if they are not in the grid-mapfile.

Verifying your Setup

- certs/keys (host and user)
  - check expiration: grid-cert-info –f /etc/grid-security/hostcert.pem –all
    - check not Before and not After
    - check –help too
  - grid-proxy-init
  - check the subject
    - the subject of the host cert should be the CA specific stuff followed by CN=host/FQDN
    - Fully qualified is important, just the hostname wont work
  - check the permissions (again)
  - grid-proxy-destroy

verifying your setup

- CA certificates
  - cd /etc/grid-security/certificates
  - All the same checks as above
  - check the hash
    - openssl x509 -hash -noout < /path/to/ca/cert
- If someone has a certificate from a CA you don't already trust
  - obtain the CA certificate and signing policy file.
  - copy them to /etc/grid-security/certificates
- grid-mapfile
  - cd ..
  - each entry on one line; DN must be in double quotes if spaces in DN

Security Environment Variables

- http://www.globus.org/security/v2.0/env_variables.html
- Note that you use X509_USER_CERT even for the server. I guess it is the user in that case (hey, I didn't come up with this stuff)

Exercise

- Examine your user security
  - NOTE: normally your userkey would have a pass-phrase associated with it
  - grid-cert-info –all
  - grid-cert-info –subject
  - grid-proxy-info –subject
    - note the difference between the cert and the proxy
  - verify your permissions (again)

Exercise

- Examine your host security
  - NOTE: normally your hostkey does NOT have a pass-phrase associated with it
    - It is protected by root read-only permissions
  - grid-cert-info –all
  - grid-cert-info –subject
  - verify your modulus
  - verify the permissions (again)

Exercise

- Check the grid-mapfile
  - cat /etc/grid-security grid-mapfile
  - grid-mapfile-add-entry –dn <make something up> -ln <account>
  - cat the file again
  - grid-mapfile-delete-entry
    - can use either –dn or –ln to specify
  - cat the file again
  - grid-mapfile-check-consistency
    - *may* be flaky in this version

Exercise

- Check the certificates directory
  - check the hash on the certificate that is there
    - openssl x509 -hash -noout < /path/to/cert.pem
  - check the expiration date
    - you can use grid-cert-info or
    - openssl x509 –dates –noout < path/to/cert.pem

Server Configuration

Server configuration

- We will take this from the web
- http://www-unix.globus.org/toolkit/docs/development/4.0-rafts/data/gridftp/GridFTP_Public_Interfaces.html#config
- Lets look at the configs on the machines

Configuration for Striping

- In reality, there is one configuration that makes something a front end (PI)
  - -r or remote_nodes
    - This causes the Master (or Remote) DSI to be loaded
    - It wont actually move things, it will just talk to the client and make IPC calls
- And there is one config that makes a back end (DTP)
  - -dn or data_node
    - causes it to start listening for IPC connections.

Configuring the logging

- Again, logging is broken in 3.9.4, but will be fixed in 3.9.5
- log_module accepts either stdio or syslog
- -Z or log_transfers puts a one entry per transfer logging all the run parameters (src, dest, user, buffer size, streams, time, etc)
- log_level you have to play with that one, I always use all ☺
- http://www-unix.globus.org/toolkit/docs/development/4.0-drafts/data/gridftp/GridFTP_Public_Interfaces.html#config

Exercise

- Work with the person next to you to set up a striped server
  - the Front End should run on 2814
  - There should be two backends
    - one on your machine on 2914
    - one on your neighbors machine on 2914
    - look at /usr/local/gridftp/gridftp[FE3|BE3] for an example
    - don't forget to add your service to /etc/services, update /etc/xinetd.d
    - restart xinetd (/etc/rc.d/init.d/xinetd restart)

Running the Server as a User

Check your build

- Hopefully, if built with no problems ☺
- In your terminal window:
  - grid-proxy-init
  - <your home>/gridftp/sbin/globus-gridftp-server –p 60000
  - grid-mapfile-add-entry –dn `globus-cert-info –subject` -ln nesc –f ~/.globus/grid-mapfile
  - use globus-url-copy as usual, but add:
    - -s `grid-proxy-info –subject`

For extra credit…

- Add your neighbors subject name to your local grid-mapfile, but map him to your local account
  - NOTE: In most real life situations, this is a definite NO-NO. You are essentially letting him use your account, which most sites have a rule against.
- Now take turns running 3rd party transfers
  - You will now have to specify the –ss and –ds seperately since one server will be running under your proxy and one will be under your neighbors

Free Time

- Feel free to play with the machine configs
- Please DO NOT mess with ports 2811,2812, 2813
- If you have ssh access to other machines, I can try and help troubleshoot your installs.